ALGEBRAIC SURFACE DESIGN AND FINITE ELEMENT MESHES

Chandrajit L. Bajaj *

Department of Computer Science,
Purdue University,
West Lafayette, IN 47907

INTRODUCTION

This paper summarizes some of the techniques used in constructing C^0 and C^1 continuous meshes of low degree, implicitly defined, algebraic surface patches in three dimensional real space $\mathbf{R}^3$. These meshes of low degree algebraic surface patches are used to construct accurate computer models of physical objects. These meshes are also utilized in the finite element simulation of physical phenomena (e.g. heat dissipation, stress/strain distributions, fluid flow characteristics, etc.) required in the computer prototyping of both the manufacturability and functionality of the geometric design.

Why use algebraic surfaces ? A real algebraic surface S in $\mathbf{R}^3$ is implicitly defined by a single polynomial equation $\mathcal{F} : f(x,y,z) = 0$, where coefficients of f are over the real numbers $\mathbf{R}$. Manipulating polynomials, as opposed to arbitrary analytic functions, is computationally more efficient. Furthermore algebraic surfaces provide enough generality to accurately model most complicated rigid objects.

Why use implicit representations ? While all real algebraic surfaces have an implicit definition $\mathcal{F}$ only a small subset of these real surfaces can also be defined parametrically by the triple $\mathcal{G}(s,t) : (x = G_1(s,t), y = G_2(s,t), z = G_3(s,t))$ where each G_i, $i = 1,2,3$, is a rational function (ratio of polynomials) in s and t over $\mathbf{R}$. The primary advantage of the implicit definition $\mathcal{F}$ is the closure properties of the class of algebraic surfaces under modeling operations such as intersection, convolution, offset, blending, etc. The strictly smaller class of parametrically defined algebraic surfaces $\mathcal{G}(s,t)$ are not closed under any of the operations listed before. Closure under modeling operations allows cascading repetitions without any need of approximation. Furthermore, designing with a larger class of surfaces leads to better possibilities of being able to satisfy the same geometric design constraints with much lower degree algebraic surfaces. The implicit representation of algebraic surfaces also naturally yields sign-invariant regions $\mathcal{F}^+ : f(x,y,z) \geq 0$ and $\mathcal{F}^- : f(x,y,z) \leq 0$, a fact quite useful for intersection and offset modeling operations. One aim of this paper is to exhibit that implicitly defined algebraic surfaces are very appropriate for constructing curved finite element meshes.

C^0 PLANAR MESHES ON RATIONAL PARAMETRIC SURFACES

It would seem that rational parametric surfaces are easy to mesh since it is trivial to generate points on a parametric surface. However in constructing a planar finite element mesh on the entire surface, two difficulties arise:

1. To mesh the entire surface one must allow the parameters to somehow range over the entire parametric domain, which is infinite. Any bounded portion of the parametric domain leaves a hole in the surface.

2. Even when restricting the surface to a bounded part of the parametric domain, the rational functions describing the surface may have poles over that domain, where the surface will become discontinous. If a surface is discontinuous, an arbitrary triangulation of the parametric domain will usually result in an incorrect triangulation of the the parametric surface.

In [13]*we give solutions to the above problems for the C^0 meshing of rational parametric curves, surfaces and hypersurfaces of any dimension [8]. The technique is based on homogeneous linear (projective) reparameterizations and yields a complete and accurate C^0 planar mesh of free-form, discontinuous rational parametric domains. For the CARTAN surface example, a single reparameterization $[x = s * t, y = s^2, z = t]$ removes the pole $t = 0$ of the original parameterization.
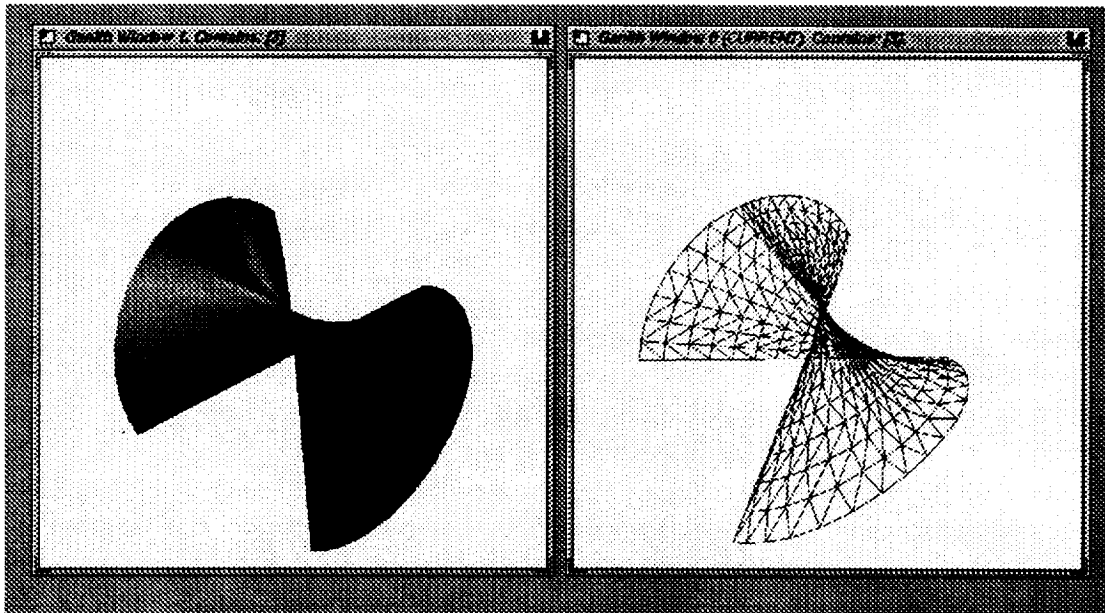

Figure 1: A finite element mesh on the CARTAN umbrella surface $[x = s, y = \frac{s^2}{t^2}, z = t]$

---

*This paper contains references 1-37.

C^0 PLANAR MESHES ON RATIONAL PARAMETRIC SURFACES, (Contd.)

For the STEINER surface example, four different projective reparameterizations yield a complete covering of the rational parametric surface. In [13] for surfaces, four reparameterizations always suffice. In general 2^d projective reparameterizations suffice for a d dimensional parametric hypersurface. The algorithms which compute these reparameterizations as well as generate the C^0 planar meshes have been implemented in C in our GANITH toolkit[12]. The pictures have been generated using this toolkit.
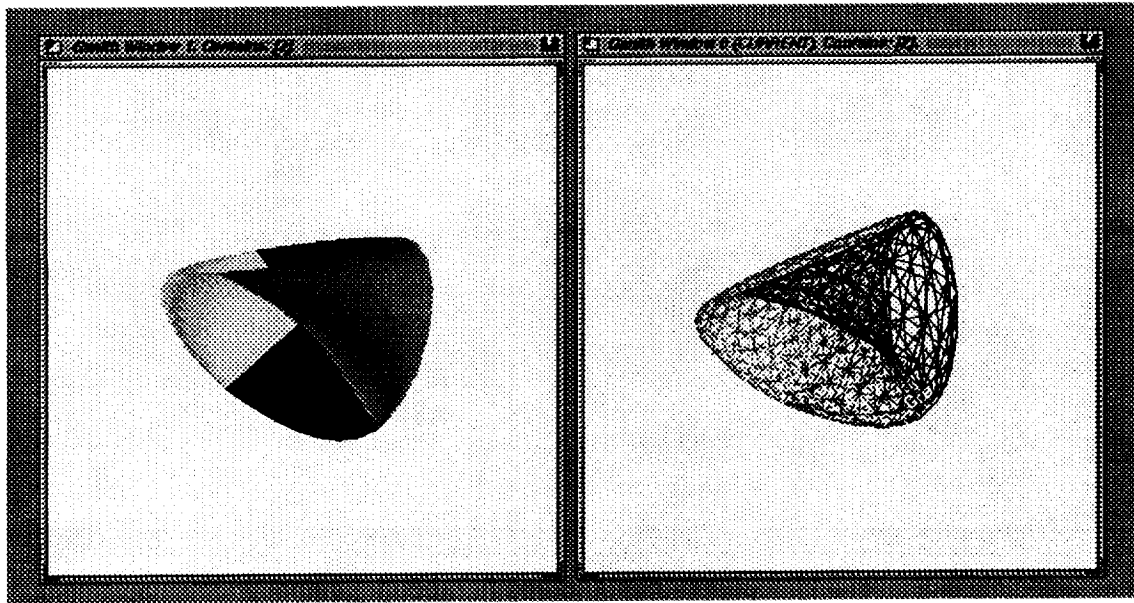

Figure 2: A finite element mesh on the STEINER surface $[x = \frac{2*s*t}{1+s^2+t^2}, y = \frac{2*s}{1+s^2+t^2}, z = \frac{2*t}{1+s^2+t^2}]$

C^0 PLANAR MESHES ON IMPLICIT ALGEBRAIC SURFACES

One way to mesh implicitly defined rational algebraic surfaces of low degree is to symbolically compute global rational parametric equations of the surface [1, 2, 8], and then use the meshing methods of the previous section. Direct schemes which work for arbitrary implicit algebraic surfaces are based on either the regular subdivision of the cube [16] or a finite subdivision of an enclosing tetrahedron [29]. The quartic surface example below was meshed in GANITH [12] by a implementation based on the methods of [16, 29].
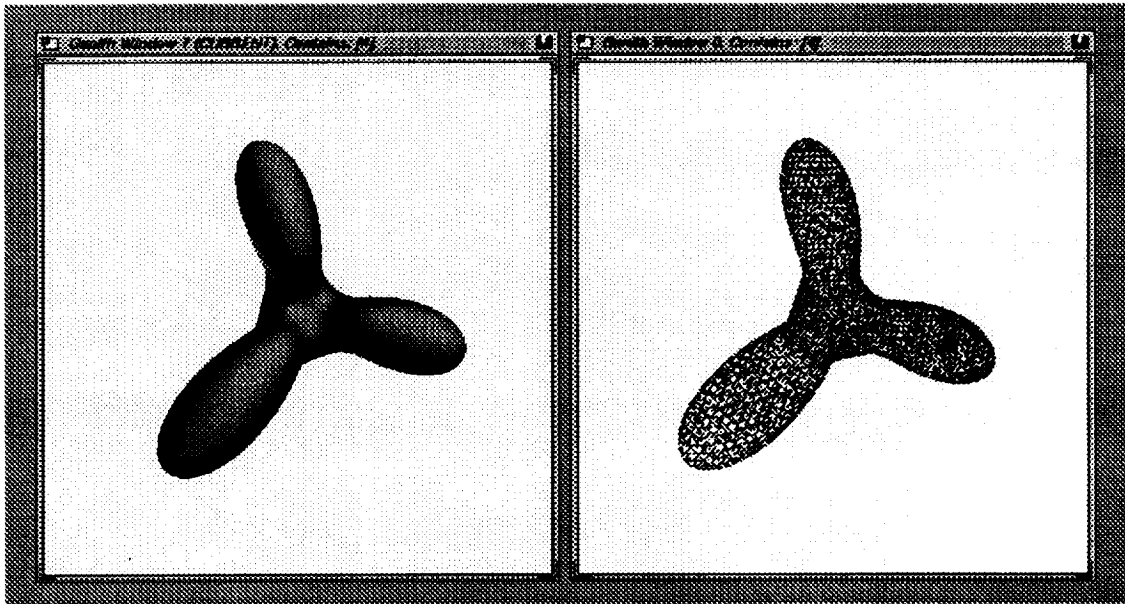

Figure 3: A finite element mesh of an implicitly defined quartic surface

C^0 PLANAR MESHES ON IMPLICIT ALGEBRAIC SURFACE PATCHES, (Contd.)

The implicitly defined triangular quintic surface patch example below was meshed in GANITH [12] by an implementation based on a space curve tracing method reported in [10]. In this method, various curves joining the boundary of the triangular patch are adaptively traced to yield a mesh of triangles which accurately C^0 approximate the triangular surface region.
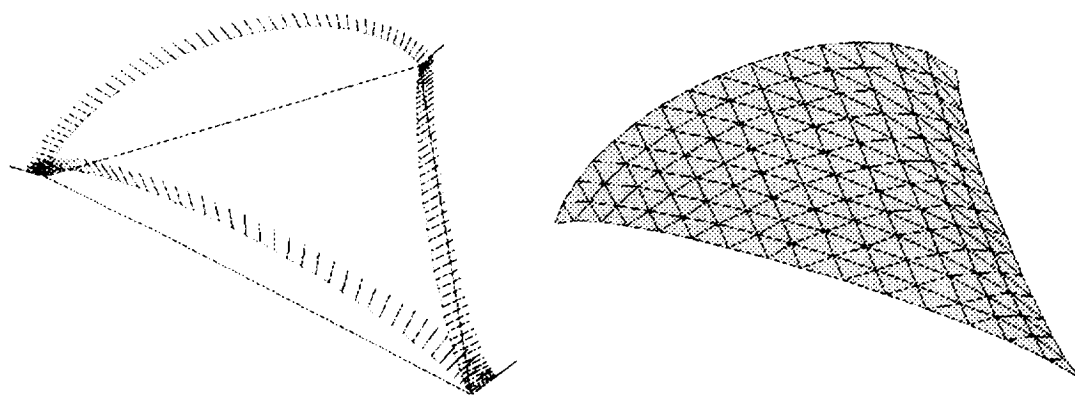

Figure 4: A finite element mesh of implicit triangular quintic patches used in C^1 smoothing polyhedra.

BOUNDED ASPECT RATIO TRIANGULATIONS IN THREE DIMENSIONS

Though a number of algorithms exist for triangulating a point set or a polyhedron in two and three dimensions [17, 23, 31], few address the problem of guaranteeing the shape of the triangular elements. To reduce ill-conditioning as well as discretization error, finite element methods require triangular meshes of bounded aspect ratio [5, 25]. By aspect ratio of triangles or tetrahedra, one may consider the ratio of the radii of the circumscribing circle to that of inscribing circle (spheres in case of tetrahedra).In 2D, two distinct approaches are known to produce guaranteed quality triangulations. The first approach, based on *Constrained Delaunay Triangulations*, was suggested by Chew [18]. He guarantees that all triangles produced in the final triangulation have angles between 30° and 120°. The other approach based on *Grid Overlaying* was used by Baker, Grosse, and Raferty in [14] to produce a non-obtuse triangulation of a planar polygon. Recently, [15], Bern, Eppstein, and Gilbert give algorithms for producing guaranteed quality triangulations. They use a regular subdivision of a square, i.e. a quadtree.

In [21] an algorithm is presented to generate good triangulations of the convex hull of a point set in three dimensions. New points are added to generate good tetrahedra with the restriction that all points are added only inside or on the boundary of the convex hull. Good triangulations of convex polyhedra are a special case of this problem. A robust implementation of this 3D triangulation algorithm, in the presence of numerical errors under finite precision arithmetic [22], has been made in our SHILP toolkit [4]. Below is one example of this implementation in SHILP.
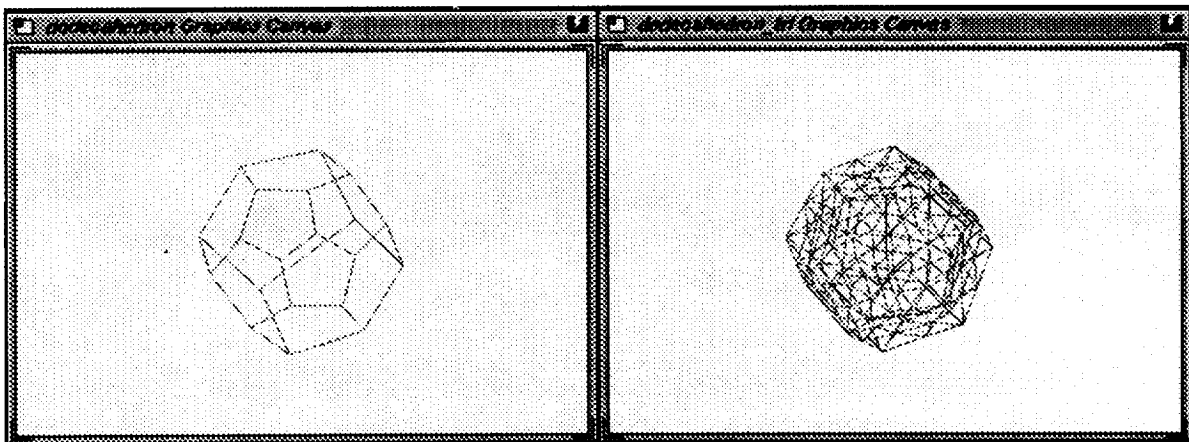

Figure 5: A finite element mesh on the surface of a dodecahedron

126

FINITE ELEMENT SOLID MODEL RECONSTRUCTION FROM CT/MRI MEDICAL IMAGING DATA

Skeletal model reconstruction from voxel data has been an active research area for many years. There are primarily two classes of model reconstruction techniques. One class of methods first constructs planar contours in each CT/NMR data slice and then connects these contours by a triangulation in three dimensional space. The triangulation process is complicated by the occurrence of multiple contours on a data slice (i.e. branching). Early contributions here are by Keppel [32], Fuchs, Kedem and Uselton [26]. The other class of methods uses a hierarchical subdivision of the voxel space to localize the triangular approximation to small cubes [33]. This method takes care of branching; however the local planar approximation based on the density values at the corner of the subcube may sometimes be ambiguous. An extension of this scheme which computes a C^1 piecewise quadratic approximation to the data within subcubes is given in [34].

In [9] we present an algorithm belonging to each of the above classes to construct C^1-smooth models of skeletal structures from CT/NMR voxel data. The boundary of the reconstructed models consists of a C^1-continuous mesh of triangular algebraic surface patches. The first algorithm starts by constructing C^1-continuous piecewise conic contours on each of the CT/NMR data slices and then uses piecewise triangular algebraic surface patches to C^1 interpolate the contours on adjacent slices. The other algorithm works directly in voxel space and replaces an initial C^0 triangular facet approximation of the model with a highly compressed C^1-continuous mesh of triangular algebraic surface patches. Both schemes are adaptive, yielding a higher density of patches in regions of higher curvature. These algorithms have been implemented in our VAIDAK toolkit [6]. An example of this reconstruction scheme from MRI data is shown below.
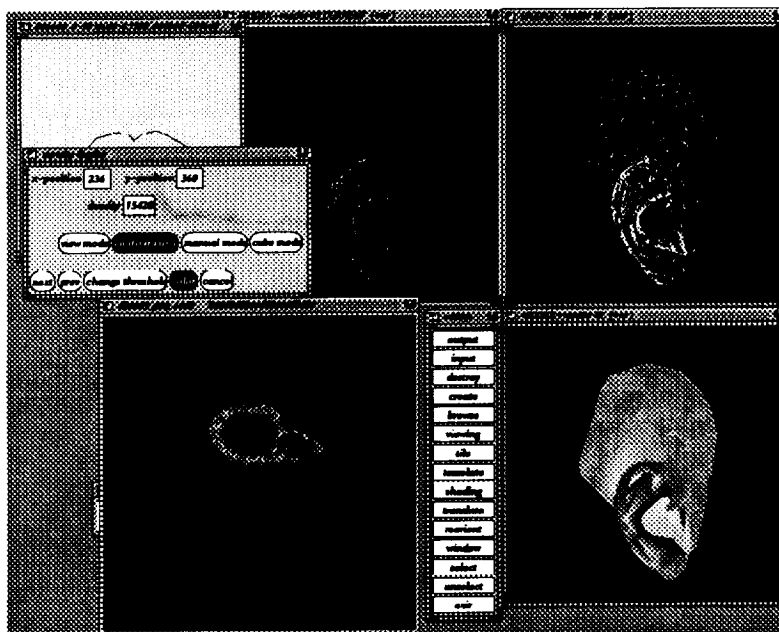

Figure 6: A finite element mesh around the ear of a human head

C^1 CURVED FINITE ELEMENT MESHES OVER TRIANGULATED DOMAINS

The generation of a C^1 mesh of smooth surface patches or *splines* that interpolate or approximate *triangulated space data* is one of the central topics of geometric design. Chui [19] summarizes much of the history of previous work. Prior work on splines has traditionally worked with a given planar triangulation using a polynomial function basis. More recently surface fitting has been considered over closed triangulation in three dimensions using a parametric surface for each triangular face [24, 27, 28, 30, 35].

Little work has been done on spline basis for implicitly defined algebraic surfaces. Sederberg [36] shows how various smooth implicit algebraic surfaces can be manipulated as functions in Bezier control tetrahedra with finite weights. Dahmen [20] presents the construction of tangent plane continuous, piecewise quadric surfaces The technique however works only if the original triangulation of the data set allows a transversal system of planes and hence is restricted. Warren [37] computes low degree blending and joining implicit surfaces by using the products of surfaces defining the blending and joining curves. Bajaj and Ihm [11] show how blending and joining algebraic surfaces can be computed via C^1 interpolation. In [10] we consider an arbitrary spatial triangulation $\mathcal{T}$ consisting of vertices (x_i, y_i, z_i) in $\mathbf{R}^3$ (or more generally a simplicial polyhedron $\mathcal{P}$ when the triangulation is closed), with possibly "normal" vectors at the vertex points. An algorithm is given to construct a C^1 continuous mesh of low degree real algebraic surface patches S_i , which respects the topology of the triangulation $\mathcal{T}$ or simplicial polyhedron $\mathcal{P}$, and C^1 interpolates all the vertices (x_j, y_j, z_j) in $\mathbf{R}^3$. The technique is completely general and uses a single implicit surface patch for each triangular face of $\mathcal{T}$ of $\mathcal{P}$, i.e. no local splitting of triangular faces. Each triangular surface patch has local degrees of freedom which are used to provide local shape control. This is achieved by use of weighted least squares approximation from points (x_k, y_k, z_k) generated locally for each triangular patch from the original patch data points and normal directions on them. These algorithms have been implemented in C in our SHILP toolkit [4]. An example is shown below.
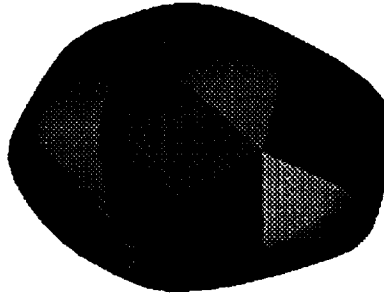

Figure 7: A C^1 curved finite element mesh over a closed spatial triangulation

References

[1] Abhyankar, S., and Bajaj, C., (1987) "Automatic Parameterization of Rational Curves and Surfaces I: Conics and Conicoids", *Computer Aided Design*, 19, 1, pp. 11 - 14.

[2] Abhyankar, S., and Bajaj, C., (1987) "Automatic Parameterization of Rational Curves and Surfaces II: Cubics and Cubicoids", *Computer Aided Design*, 19, 9, pp. 499 - 502.

[3] Alfeld, P., (1989) "Scattered Data Interpolation in Three or More Variables", *Mathematical Methods in Computer Aided Geometric Design*, ed. Lyche, T. and Schumaker, L., pp. 1-34.

[4] Anupam, V., Bajaj, C., Dey, T., Ihm, I., (1991), "The SHILP Solid Modeling and Display Toolkit", Computer Science Tech. Report, CSD-TR-91-064, CAPO-91-29, Purdue University.

[5] Babuska and Aziz, A., (1976), "On the Angle Condition in the Finite Element Method", *SIAM Numerical Analysis*, 13, 214-226.

[6] Bailey, B., Bajaj, C., Fields, M., (1991), "The VAIDAK Medical Imaging and Model Reconstruction Toolkit", Computer Science Tech. Report, CSD-TR-91-066, CAPO-91-31, Purdue University.

[7] Bajaj, C., (1989) "Geometric Modeling with Algebraic Surfaces" *The Mathematics of Surfaces III*, ed., D. Handscomb, Oxford University Press, pp. 3 - 48.

[8] Bajaj, C., (1990) "Rational Hypersurface Display", Proc. of 1990 Symposium on Interactive 3D Graphics, *Computer Graphics*, 24, 2, pp. 117 - 128.

[9] Bajaj, C., (1991) "Electronic Skeletons: Modeling Skeletal Structures with Piecewise Algebraic Surfaces", in Curves and Surfaces in Computer Graphics and Vision II, *Proc. of the Symposium on Electronic Imaging Science & Technology*, Boston, MA, vol. 1610, pp. 231 - 237.

[10] Bajaj, C., and Ihm, I., (1991), "The C^1 Smoothing of Polyhedra with Implicit Surface Patches", Computer Science Tech. Report, CSD-TR-91-060, CAPO-91-27, Purdue University.

[11] Bajaj, C. and Ihm, I., (1992), "Algebraic Surface Design with Hermite Interpolation" *ACM Transactions on Graphics*, 19, 1, (1992).

[12] Bajaj, C., and Royappa, A., (1991), "The GANITH Algebraic Geometry Toolkit, V1.0", Computer Science Tech. Report, CSD-TR-91-065, CAPO-91-30, Purdue University.

[13] Bajaj, C., and Royappa, A., (1992), "The Robust Display of Free-Form Rational Parametric Surfaces", Computer Science Tech. Report, CSD-TR-92-005, CAPO-92-04, Purdue University.

[14] Baker, B., Grosse, E., and Rafferty, C., (1988), "Nonobtuse Triangulation of Polygons", *Discrete and Computational Geometry*, 3, pp. 147-168.

[15] Bern, M., Eppstein, D., and Gilbert, J., (1990), "Probably Good Mesh Generation", *Proc. 31st Annual IEEE Symposium on Foundations of Computer Science*, pp. 231-241.

[16] Bloomenthal, J., (1988), "Polygonization of implicit surfaces", *Computer Aided Geometric Design*, vol. 5, pp. 341-355.

[17] Chazelle, B., and Palios, L., (1990), "Triangulating a Non-convex Polytope", *Discrete and Computational Geometry*, 5, pp. 505-526.

[18] Chew, L. P., (1989), "Guaranteed-Quality Triangular Meshes", Technical Report TR-89-983, Cornell University.

[19] Chui, C., (1988), *Multivariate Splines*, Regional Conference Series in Applied Mathematics, SIAM.

[20] Dahmen, W., (1989) "Smooth Piecewise Quadric Surfaces", *Mathematical Methods in Computer Aided Geometric Design*, ed. Lyche, T. and Schumaker, L., pp. 181-194.

[21] Dey, T., Bajaj, C., and Sugihara, K., (1991) "On Good Triangulations in Three Dimensions", **Proc. of the ACM Symposium on Solid Modeling Foundations and CAD/CAM Applications**, (1991), pp. 131-141.

[22] Dey, T., Sugihara, K., and Bajaj, C., (1992) "Triangulations in Three Dimensions with Finite Precision Arithmetic", Computer Science Tech. Report, CSD-TR-92-001, CAPO-92-01, Purdue University.

[23] Edelsbrunner, H., Preparata, F., and West, D., (1986), "Tetrahedrizing Point Sets in Three Dimensions", Tech. Report UIUCDCS-R-86-1310.

[24] Farin, G., (1986), "Triangular Bernstein-Bezier Patches", *Computer Aided Geometric Design*, 3, pp. 83 - 127.

[25] Fried, I., (1972), "Condition of Finite Element Matrices Generated from Nonuniform Meshes", *AIAA J.*, 10, pp. 219-221.

[26] Fuchs, H., and Kedem, Z., and Uselton, S., (1977), "Optimal Surface Reconstruction from Planar Contours", *Communications of the ACM*, vol. 20, pp. 693-702.

[27] Gregory, J., and Charrot, P., (1980) "A C^1 Triangular Interpolation Patch for Computer Aided Geometric Design" *Computer Graphics and Image Processing*, 13, pp. 80-87.

[28] Hagen, H. and Pottmann, H., (1989) "Curvature Continuous Triangular Interpolants", *Mathematical Methods in Computer Aided Geometric Design*, ed. Lyche, T. and Schumaker, L., pp. 373-384.

[29] Hall, M., and Warren, J., (1990), "Adaptive Polygonalization of Implicitly Defined Surfaces", *IEEE Computer Graphics and Applications*, pp. 33-42.

[30] Herron, G., (1985) "Smooth Closed Surfaces with Discrete Triangular Interpolants", *Computer Aided Geometric Design*, 2, 4, pp. 297-306.

[31] Joe, B., (1989), "Three-dimensional Triangulations from Local Transformations", *SIAM J. Sci. Stat. Comput.*, 10, pp. 718-741.

[32] Keppel, E., (1975), "Approximating Complex Surfaces by Triangulation of Contour Lines", *IBM J. Research and Development*, vol. 19, pp. 2-11.

130

[33] Lorensen, W. and Cline, H., (1987), "Marching Cubes: A High Resolution 3D Surface Construction Algorithm", *Computer Graphics*, pp. 163-169.

[34] Moore, D., and Warren, J., (1991) "Approximation of Dense Scattered Data using Algebraic Surfaces", *Proc. of the 24th Hawaii Intl. Conference on System Sciences*, Kauai, Hawaii, pp. 681-690.

[35] Peters, J., (1990) "Smooth Mesh Interpolation with Cubic Patches" *Computer Aided Design*, 22, 2, pp. 109-120.

[36] Sederberg, T., (1985), "Piecewise Algebraic Surface Patches", *Computer Aided Geometric Design*, 2, pp. 53-59.

[37] Warren, J., (1989), "Blending Algebraic Surfaces", *ACM Transactions on Graphics*, vol. 8, no. 4, pp. 263-278.